Multi-level Contrastive Learning on Weak Social Networks for Information Diffusion Prediction

Zihan Feng[1], Rui Wu[1], Yajun Yang[1(✉)], Hong Gao[2], Xin Wang[1], Xueli Liu[1], and Qinghua Hu[1]

[1] College of Intelligence and Computing, Tianjin University, Tianjin, China
{zihanfeng,circle,yjyang,wangx,xueli,huqinghua}@tju.edu.cn
[2] College of Mathematics and Computer Science, Zhejiang Normal University, Jinhua, China
honggao@zjnu.edu.cn

Abstract. Information diffusion prediction, as a fundamental task in social network analysis, aims to identify potential users who are likely to participate in an information diffusion process. Most existing works learn user representations based on the collected social network data and then complete downstream prediction tasks. However, due to data privacy protection and low data quality, these methods are always limited by weak information issues of the social network data. For example, incomplete network structure, sparse labels, and insufficient features severely obstruct user representation learning. To mitigate these issues, we design an effective two-stage method MGCL. In the first stage, an enhanced representation is learned for every user even though the social network is with weak information. A multiplex heterogeneous network is adaptively constructed to enrich social network information. To facilitate user representation learning under sparse labels and insufficient features, we further propose self-supervised training specifically tailored for social networks with weak information. In the second stage, the cascade representations are learned using the multi-head self-attention network for information diffusion prediction. Extensive experiments on four real-world datasets validate that MGCL always outperforms state-of-the-art methods.

Keywords: Social networks analysis · Information diffusion prediction · Graph representation learning · Contrastive learning

1 Introduction

With the rapid development of the mobile Internet, social network platforms have become the most important place for people to obtain and share information. Information diffusion prediction is a fundamental task in social networks, which can assist in understanding the evolution of trending topics and has many

M. Onizuka et al. (Eds.): DASFAA 2024, LNCS 14855, pp. 84–100, 2024.
https://doi.org/10.1007/978-981-97-5572-1_6

important applications, such as public opinion analysis [1], fake news control [2], and online social marketing [3].

Information diffusion prediction is to identify potential users who are likely to participate in information sharing, e.g., re-tweets. Existing methods can be classified into two categories: predefined model-based methods and cascade learning-based methods. Predefined model-based methods artificially design various propagation models [4] (e.g., SIR model) to predict which users will be involved in the information diffusion. However, these methods may not be strictly consistent with information propagation in the real world. To address this limitation, cascade learning-based methods have been proposed recently. An information diffusion cascade is a sequence of tuples (v_x, t_x), where v_x is a user who participates in this propagation and t_x is her/his participation time. On the one hand, some works focus on extracting the macro-level features, such as user statistics or information contents, to predict the overall growth of cascade size [5,6]. On the other hand, some methods aim to predict the next user that will participate in this cascade based on representation learning. These methods utilize various techniques such as GNNs and RNNs to learn users' social relationships and sequential cascade interaction [10–13] for information diffusion prediction.

Despite the improvements made in describing the user social homogeneity and cascade interaction, existing methods still suffer from three critical limitations: (1) Existing works are designed on the fundamental assumption that observed social network data contains complete and sufficient information. However, in the real world, these data suffer from extremely weak information issues, manifested by incomplete structure, sparse labels, and insufficient user features. Such low-quality data severely obstructs learning user representations. (2) All existing methods only utilize cascade sequences as the prediction-oriented objective to jointly learn social networks and information diffusion cascades, which is insufficient for learning reliable user representations of the social network and further results in suboptimal prediction performance. (3) Existing end-to-end methods all use the full-graph training pipeline for the entire social network, which is straightforward but significantly restricts practical flexibility.

To address these limitations, we propose a novel two-stage method MGCL for information diffusion prediction. In the first stage, an enhanced representation can be learned for every user even though the social network only has weak information. On the one hand, by leveraging the relationships of users' follower-followee and historical participation behavior, a multiplex heterogeneous social network is constructed to enrich the structure information in the original social network and then an adaptive early-fusion strategy is well-designed to ensure a reliable adjacency matrix for graph learning. On the other hand, to facilitate user representation learning under sparse labels and insufficient features, we propose self-supervised training specifically tailored for social networks with weak information and further design contrastive learning tasks at the user and preference level respectively. Our method can effectively mitigate the challenges posed by weak information issues to user preference modeling and thereby improve the performance of downstream prediction. Moreover, we employ an inductive graph

learning method with the mini-batch setting that can cope with the graph structure changing and is more flexible for model training. In the second stage, the cascade representations are learned using the multi-head self-attention network for the downstream information diffusion prediction task.

The main contributions of this paper are summarized as follows:

(1) We propose an effective method to enhance the original social network by integrating users' historical behavior into a multiplex heterogeneous social network and designing an adaptive early-fusion strategy to ensure a reliable user adjacency matrix for graph learning.
(2) We design a novel multi-level graph contrastive learning method that can provide more sufficient self-supervised signals for graph learning. In this way, our method can obtain more expressive user representations even if it does not have user features and labels.
(3) We conduct extensive experiments on four real-world datasets. The results show that: (i) MGCL can enable more effective supervision for graph learning and further effectively improve prediction performance. (ii) In various scenarios, MGCL consistently outperforms the state-of-the-art methods.

2 Preliminaries

Definition 1. *(***Information Diffusion Cascade***). An information diffusion cascade C_k with k users is a sequence of k tuples, i.e., $C_k = ((v_1, t_1), (v_2, t_2), \ldots, (v_k, t_k))$, where $v_x \neq v_y$ and $t_x \leq t_{x+1}$. Each tuple (v_x, t_x) is called a "participation" indicating that user v_x is the x-th participant in information diffusion and the participation time is t_x.*

The information diffusion process can be regarded as a communication chain. An information diffusion cascade records the complete journey of a specific piece of information spreading through the social network. In information diffusion prediction, C_k is used as an input to predict the next participation (v_{k+1}, t_{k+1}).

Definition 2. *(***Ideal Social Network***). We define the ideal social network as social network data that contains sufficient information. Let the ideal social network be $\hat{\mathcal{D}} = (\hat{\mathcal{G}}, \hat{\mathbf{Y}}_L) = ((\mathcal{V}, \hat{\mathcal{E}}, \hat{\mathbf{X}}), \hat{\mathbf{Y}}_L)$, where $\hat{\mathcal{E}}$ is an ideal edge set containing all relevant links between users, $\hat{\mathbf{X}}$ is an ideal feature matrix containing all informative features for each user, and $\hat{\mathbf{Y}}_L$ is an ideal label matrix containing sufficient labels (with number $\hat{n}_L$) for learning tasks with balanced distribution.*

In real-world scenarios, there often is insufficient data for model training and deployment [14]. Specifically, the structure can be incomplete with an *incomplete edge set* $\check{\mathcal{E}} \subsetneqq \hat{\mathcal{E}}$ that contains limited edges to provide sufficient information. Meanwhile, some critical elements in the feature matrix are missing, which can be represented by an *incomplete feature matrix* $\check{\mathbf{X}} = \mathbf{M} \odot \hat{\mathbf{X}}$, where $\mathbf{M} \in \{0, 1\}^{n \times d}$ is the missing mask matrix. Besides, the available labels for training can be scarce, resulting in an *insufficient label matrix* $\check{\mathbf{Y}}_L$ with training number $\check{n}_L \ll \hat{n}_L$.

Definition 3. *(***Weak Social Network***). Due to privacy security issues, weak social networks usually do not have user features and labels, and observed relationships are always incomplete, further exacerbating the challenges. Let social network data with weak information be* $\mathcal{D}_x = ((\mathcal{V}, \check{\mathcal{E}}, \check{\mathbf{X}}), \check{\mathbf{Y}}_L)$*. The target is to learn the representation of users* $\mathcal{V}$ *with* $\mathcal{D}_x$ *for model training.*

In this paper, we confront a more challenging scenario wherein the structure, features, and labels exhibit serious deficiencies. Specifically, manual labels are absent for all users, initial user features are randomized, and observed user relationships are extremely incomplete. The sole additional information available is users' historical participation in information diffusion processes.

Definition 4. *(***Multiplex Heterogeneous Network***). A multiplex heterogeneous network can be denoted as* $G = (V, E)$*, where* V *and* E *are the sets of nodes and edges respectively.* $\phi : V \rightarrow O$ *is a node type mapping function and* $\psi : E \rightarrow R$ *is an edge type mapping function, where* O *and* R *represent the sets of node types and edge types respectively, and they satisfy* $|O| + |R| > 2, |R| > 1$*. Different from traditional heterogeneous networks, there exist multiple edges with different types between two nodes, i.e.* $E \subseteq V \times V \times R$*.*

In the real world, a social network inherently involves multiple types of complex relationships between users and thus it can be modeled by a multiplex heterogeneous network. For instance, follower-followee relationships on platforms such as Twitter constitute one type of edge. Moreover, users engaging in the same information diffusion events can be viewed as another type of edge, signifying shared interests in specific trending topics. This multiplex heterogeneous network, comprising multiple types of relationships, can offer more comprehensive information conducive to mitigating weak information issues.

Problem Formulation. The task of information diffusion prediction is to predict the next participation (v_{k+1}, t_{k+1}) based on the current information diffusion cascade C_k. Formally, the prediction task is to select an optimal v_x from $V \backslash V_k$, to maximize the following conditional likelihood, as the next participant v_{k+1}.

$$\hat{v}_x = \arg\max P(v_x | G, C_k), v_x \in V \backslash V_k \tag{1}$$

where V_k is the set of vertices in C_k, G is the social network and C_k is the information diffusion cascade sequence.

3 Methodology

Overview. The overall architecture of MGCL is shown in Fig. 1. Specifically, MGCL is a two-stage model designed for the information diffusion prediction task. In the first stage, a multiplex heterogeneous social network is constructed adaptively, taking into account both the original relationships of users' follower-followee connections and their historical participation. This enhanced structure

mitigates weak information issues present in the original social network. Subsequently, we design three self-supervised tasks that can contribute to adaptive graph construction and effective graph learning. In this way, we can obtain better user representations for the downstream prediction even if it does not have user features and labels. In the second stage, we further exploit the multi-head self-attention network to learn the information diffusion cascade for prediction.

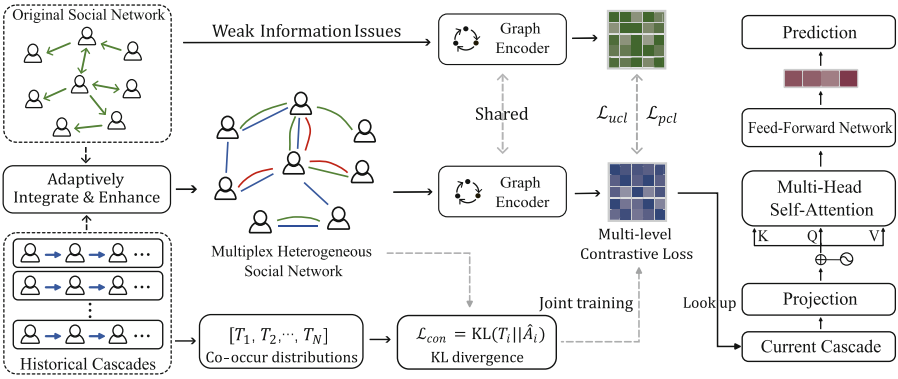


Fig. 1. The overall architecture of the proposed method MGCL.

3.1 Multiplex Heterogeneous Graph Learning

Adaptive Graph Construction. An ideal graph structure is the basis for graph learning. However, the weak social network hinders the graph encoder's ability to capture crucial dependencies between users and leads to biased user representations [14]. By leveraging the assumption that user dependencies mirror real-world information diffusion patterns, we propose an adaptive graph construction method to enhance the original social network. On the one hand, users' historical participation relationship indicates which users prefer to participate in the same information diffusion and prediction task can benefit from it. Given an information diffusion cascade C_k and an integer h, an edge (v_x, v_{x+i}) will be built for every $i \leq h$ and every v_x and v_{x+i} in C_k. For example of $h = 3$, three edges (v_x, v_{x+1}), (v_x, v_{x+2}), and (v_x, v_{x+3}) are created for every v_x in C_k, where $1 \leq \mathrm{x} \leq k-3$. We can select an appropriate h to state users' historical participation. On the other hand, the follower-followee relationships can be treated as two types of edges due to the limitation of graph convolution on directed graphs. For example, user v_y is a follower of user v_x, thus (v_y, v_x) and (v_x, v_y) are created to represent the follower and followee relationship respectively. The intuitive meaning behind that is followers are usually influenced by their followees but not vice versa and it reveals the different influential roles of users.

For each edge type $r \in R$, a "***projection***" of a multiplex heterogeneous network G on edge type r, denoted as $G_r = (V, E_r)$, is a subgraph of G consisting

of all the nodes in V and all the edges in E_r, where $E_r = \{e|e \in E \wedge \psi(e) = r\}$. Each G_r $(1 \leq r \leq |R|)$ only focuses on one type of relationship in G. We use $\boldsymbol{A_r}$ to denote the adjacency matrix of G_r. To adaptively integrate and enhance multiplex heterogeneous relationships between users, we utilize the relation-aware weights β_r to aggregate the adjacency matrix of different projections as follows:

$$\hat{\boldsymbol{A}} = \sum_{r=1}^{|R|} \beta_r \boldsymbol{A_r} \tag{2}$$

where β_r $(1 \leq r \leq |R|)$ is a learnable relationship weight. We use the *softmax* function to ensure $\sum_{r=1}^{|R|} \beta_r = 1$. In this way, the weights imply the importance of different types of relationships, and thus the aggregated adjacency matrix $\hat{\boldsymbol{A}}$ can characterize the multiplex heterogeneous relationships among users.

Inductive Graph Encoder. In this paper, we use the inductive GraphSAGE [16] as the basic graph encoder to propagate the node message exploiting the multiplex heterogeneous social network $\hat{\boldsymbol{A}}$. The message propagation and aggregation at the t-th layer of the graph neural network are as follows:

$$\alpha_{v_i}^{(t)} = \text{Aggregate}^{(t)}(\{h_{v_j}^{(t-1)} : v_j \in N_{v_i}\}) \tag{3}$$

$$h_{v_i}^{(t)} = \text{Update}^{(t)}(\alpha_{v_i}^{(t)}, h_{v_i}^{(t-1)}) \tag{4}$$

where N_{v_i} denotes the set of v_i's neighbors in $\hat{\boldsymbol{A}}$, $h_{v_i}^{(t)}$ denotes the representation of v_i at the t-th GNN layer. $h_{v_i}^{(0)}$ is the initial user embedding randomly generated by the normal distribution. Aggregate$(\cdot)$ is the function aggregating the neighborhood information of a central node v_i, and Update$(\cdot)$ is the function that combines neighborhood information to update the node embedding. In this paper, we use weighted aggregation as the neighborhood aggregation function based on the weight of relationships in the adjacency matrix $\hat{\boldsymbol{A}}$. We denote the embedding at the last layer as the final user representation, which can be used as an input for subsequent information diffusion prediction.

3.2 Self-supervised Graph Training

Previous works entangle graph training and the downstream prediction task for end-to-end learning, but this joint training pipeline has two serious limitations: (1) The supervision signal of joint training predominantly focuses on downstream prediction, overlooking crucial supervision for the social network itself. (2) The joint training pipeline necessitates preloading the entire social network to acquire extensive representations of all users and subsequently perform lookups during downstream tasks. This unnecessary overhead constrains the practical flexibility of GNN-based methods. To achieve more efficient and effective graph learning, we design mini-batch self-supervised objectives to supervise graph construction and user representation learning. Subsequently, these representations can be projected into the downstream model for improved prediction.

Graph Structure Regularization. A critical foundation of graph learning is to generate a more reliable adjacency matrix $\hat{\boldsymbol{A}}$ that requires effective learning of β_r in Eq. (2). However, without proper supervision, the adaptive strategy may result in sub-optimal solutions, i.e., the quality of the constructed graph may not be consistent with the preferences exhibited by users in actual information diffusion participation. Inspired by [18], we leverage co-occur patterns in historical cascades as supervision to conduct graph structure regularization. This method capitalizes on the reasonable observation that frequently co-occurring users in historical cascades often share similar preferences, making it a natural and effective supervision for constructing a reliable graph structure.

We first precalculate the co-occur pattern $P_i \in N^N$ for each user v_i, where the co-occur weight to the j-th user equals the total frequency that v_i and v_j participate in the same information diffusion in all historical cascades. Furthermore, we transform P_i to the discrete distribution $T_i \in N^N$ by row normalization and softmax operation. We formalize the regularization of graph structure as the KL divergence between the approximate distribution of each user from the adjacency matrix $\hat{\boldsymbol{A}}$ and its corresponding distribution T_i of co-occur regularity, which indicates the discrepancy between these two distributions. In practice, we transform each row $\hat{A}_i$ of $\hat{\boldsymbol{A}}$ to the approximate distribution with softmax operation and compute the regularization term $\mathcal{L}_{con}$ as:

$$\mathcal{L}_{con} = \frac{1}{N} \sum_{i=1}^{N} KL(T_i || softmax(\hat{A}_i)) \tag{5}$$

This graph structure regularization objective ensures a refined graph structure, consequently mitigating the challenges posed by weak information issues. Moreover, the learned graph structure $\hat{\boldsymbol{A}}$ explicitly captures the pertinent dependencies between users in real-world scenarios, which is more beneficial for prediction.

Multi-level Contrastive Learning. Then we can conduct graph encoder training on the constructed multiplex heterogeneous social network. Motivated by the principle that user representations should be consistent with their inherent preferences in information diffusion, we introduce a novel multi-level graph contrastive learning method with the mini-batch setting, which operates on two different levels: user-level and preference-level.

As shown in Fig. 2, denote the distance between the representations of v_i in different views as d_u, the distance between v_i and another user v_j with similar preferences as d_s, and the distance between the user and another user v_k with quite different preference as d_f. Note that the generated user representations can be different due to input graph structure differences, i.e. $0 < d_u$. Meanwhile, excessively increasing d_u will result in overlapping between users with the other and therefore lead to suboptimal results, i.e. $d_u < \{d_s, d_f\}$. Furthermore, user representations with similar preferences should have smaller distances than those with larger differences, i.e. $d_s < d_f$. To further refine our distance considerations, we introduce d_δ representing the approximate boundary of the preference decision area of v_i. Ideally, the representation of user v_i should be distributed at

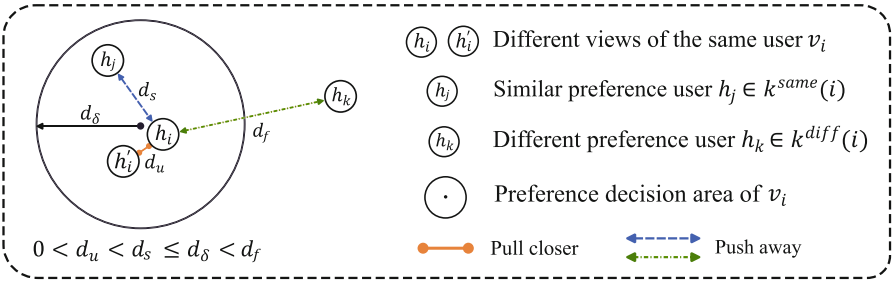


Fig. 2. Here is an example of multi-level contrastive learning.

the center of the area, and then d_s can be approximated as the distance from h_j to the center of the area. Therefore, we strive to maintain $d_s \leq d_\delta$ (or $d_s \approx d_\delta$), ensuring h_j resides within the appropriate boundary.

In summary, the total goal can be further described as:

$$0 < d_u < d_s \leq d_\delta < d_f \tag{6}$$

Based on the guidance of this goal, we can generate positive samples and negative samples within a mini-batch. Then we take user v_i as an example to introduce the detailed generation strategy. Given the representation h_i of user v_i learned from the constructed multiplex heterogeneous social network as a ***query***.

Positive Key Generation. In Sect. 3.1, we enhance the original social network by integrating the original structure and historical participation. Although the graph structure is not the same, the user representations in the original social network and the enhanced social network should be similar. Therefore, we also learn the original social network using the shared graph encoder. In this way, we can obtain the original user representation h_i^* as the positive key.

Negative Key Generation. Initially, we treat all representations of other in-batch samples as its negative keys, where $k^{neg}(i) = \{h_j, h_j^*\}_{j=1\ldots n, j\neq i}$. However, this does not represent the crucial link between user representations and their actual information diffusion behavior. Furthermore, $k^{neg}(i)$ can be divided into two groups according to the co-occur pattern $P_i \in N^N$:

$$k^{diff}(i) = \{k \in k^{neg}(i) : P_{i,k} < \delta\} \tag{7}$$

$$k^{same}(i) = \{k \in k^{neg}(i) : P_{i,k} \geq \delta\} \tag{8}$$

where $k^{diff}(i)$ is the group of users who do not frequently co-occur with user v_i in this batch, and $k^{same}(i)$ is the opposite. In this way, these two groups of users can respectively represent two types of users with different and similar preferences as user v_i in information diffusion. This in-batch generation of positive and negative examples ensures the scalability of our method. Now given the query h_i with its positive key h_i^* and negative keys $k^{neg}(i) = \{k^{diff}(i), k^{same}(i)\}$, we can introduce the user-level and preference-level contrastive loss as follows.

User-level Contrastive Loss. We aim to guarantee $0 < d_u < \{d_s, d_f\}$ at the user level. To achieve this, we design an InfoNCE-based loss as follows:

$$\mathcal{L}_{ucl} = -\frac{1}{N}\sum_{i=1}^{N}\left(\log\frac{\mathrm{f}(h_i, h_i^*)}{\mathrm{f}(h_i, h_i^*) + \sum_{j\neq i}\mathrm{f}(h_i, h_j^*)}\right) \tag{9}$$

where $\mathrm{f}(\mathbf{a}, \mathbf{b}) = \mathrm{e}^{\mathrm{sim}(\mathbf{a},\mathbf{b})/\tau}$, $\mathrm{sim}(\cdot,\cdot)$ is the cosine similarity, and τ is temperature coefficient. Note that taking $\mathrm{f}(h_i, h_i^*)$ as the denominator can ensure $0 < d_u$.

Preference-level Contrastive Loss. At the preference level, our goal is to satisfy $0 < \{d_s, d_\delta\} < d_f$. If depict the exact preference boundary of user v_i, we should calculate for each user in $k^{same}(i)$, but this is quite time-consuming. Therefore, we introduce the preference prototype. For each user v_i, we acquire its preference prototype $p_i, p_i^* = \mathbf{MEAN}(k^{same}(i))$, which are correlated to the preference boundary d_δ. The preference-level contrastive loss can be defined as:

$$\mathcal{L}_{pcl} == -\frac{1}{N}\sum_{i=1}^{N}\left(\log\frac{\mathrm{f}(h_i, p_i)}{\sum_{j\in k^{diff}(i)}\mathrm{f}(h_i, h_j)} + \log\frac{\mathrm{f}(h_i, p_i^*)}{\sum_{j\in k^{diff}(i)}\mathrm{f}(h_i, h_j^*)}\right) \tag{10}$$

where $\mathrm{f}(\mathbf{a}, \mathbf{b})$ is the same as Eq.(9). $\mathcal{L}_{pcl}$ increases the agreement between the user and its similar users, which explicitly incorporates the crucial link between user representations and their actual information diffusion behavior.

Overall Training Objective. *Graph Structure Regularization* and *Multi-level Contrastive Learning* are all formulated for a single node, eliminating the need to preload the entire graph. So the overall graph training objective is:

$$\mathcal{L}_{graph} = \gamma_1\mathcal{L}_{con} + \gamma_2\mathcal{L}_{ucl} + \gamma_3\mathcal{L}_{pcl} \tag{11}$$

where $\gamma_1, \gamma_2, \gamma_3$ are the hyper-parameter. In each mini-batch, we randomly sample n nodes to calculate $\mathcal{L}_{graph}$. In this way, we can simultaneously achieve better graph construction and better graph learning, which provides an effective and comprehensive solution for learning the weak social network.

3.3 Information Diffusion Prediction

In this section, we aim to convert the user representation sequence into the cascade representation for information diffusion prediction. Here we employ the basic multi-head self-attention mechanism [17] and eschew more complex encoders, which emphasizes the effectiveness of our proposed graph learning method.

Cascade Interaction Encoder. We first look up user representations $\boldsymbol{H_k} = \{\boldsymbol{h_x}|v_x \in V_k\}$ from multiplex heterogeneous graph encoder for the cascade C_k. Then we encode position information to obtain $\boldsymbol{H_k^{'}} = \{\boldsymbol{h_x^{'}}|v_x \in V_k\}$, where $\boldsymbol{h_x^{'}} =$

$[\boldsymbol{h_x}||\boldsymbol{p_x}]$ and the $\boldsymbol{p_x}$ is a learnable position embedding. $\boldsymbol{H_k^{'}}$ is user representations in C_k and preserves the relative sequential information of users.

We further employ the basic Multi-Head Attention (MHA) to model cascade interaction. The MHA of the i-th attention head can be calculated as follows:

$$\boldsymbol{Q_i} = \boldsymbol{H_k^{'}}\boldsymbol{W_i^Q}, \boldsymbol{K_i} = \boldsymbol{H_k^{'}}\boldsymbol{W_i^K}, \boldsymbol{V_i} = \boldsymbol{H_k^{'}}\boldsymbol{W_i^V} \tag{12}$$

$$\mathrm{MHA}_i(\boldsymbol{Q_i}, \boldsymbol{K_i}, \boldsymbol{V_i}) = \mathrm{softmax}(\frac{\boldsymbol{Q_i}\boldsymbol{K_i}^{\mathsf{T}}}{\sqrt{d^{'}}} + \boldsymbol{M})\boldsymbol{V_i} \tag{13}$$

where d is the dimension of the embedding, B represents the number of heads, $d^{'} = d/B$, $\boldsymbol{W_i^Q}$, $\boldsymbol{W_i^K}$, and $\boldsymbol{W_i^V}$ are learnable projection matrices. To avoid label leakage, we introduce a mask matrix $\boldsymbol{M} \in \mathbb{R}^{n \times n}$ to block out future information and achieve this via setting the upper triangle of the attention map with $-\infty$. We concatenate the outputs of B attention heads to form the output $\boldsymbol{S}$ as:

$$\boldsymbol{S} = [\mathrm{MHA}_1|| \ldots ||\mathrm{MHA}_B]\boldsymbol{W^O} \tag{14}$$

where $||$ is the concatenation operation, and $\boldsymbol{W^O}$ is the transformation matrix.

To introduce non-linearity to the MHA, we use a position-wise feed-forward network, allowing it to model complex dependencies within the cascade:

$$\boldsymbol{Z} = (\mathrm{ReLU}(\boldsymbol{S}\boldsymbol{W_t^1} + \boldsymbol{b_t^1}))\boldsymbol{W_t^2} + \boldsymbol{b_t^2} \tag{15}$$

where $\boldsymbol{W_t^1}$, $\boldsymbol{W_t^2}$, $\boldsymbol{b_t^1}$, and $\boldsymbol{b_t^2}$ are learnable parameters, $\boldsymbol{Z} \in \mathbb{R}^{k \times d}$ is the final information diffusion cascade representation.

Prediction. After that, we calculate the probability $\hat{y} \in \mathbb{R}^{|C_k| \times |V|}$ of the next participant using a softmax function:

$$\hat{y} = \mathrm{softmax}(\boldsymbol{W_p}\boldsymbol{Z} + Mask) \tag{16}$$

where $\boldsymbol{W_p}$ maps $\boldsymbol{Z}$ to user-specific space, $Mask$ is used to mask users who have participated in information diffusion before prediction. Finally, we can use the cross-entropy function to supervise the training of cascade prediction:

$$\mathcal{L}(\theta) = -\sum_{j=2}^{|C_k|}\sum_{i=1}^{|V|} y_{ij}\log(\hat{y}_{ij}) \tag{17}$$

in which θ denotes all parameters needed to be learned in the model, if the user v_i participates in cascade C_k at the step j, $y_{ij} = 1$, otherwise $y_{ij} = 0$.

4 Performance Evaluation

In this section, we conduct extensive experiments to evaluate the performance of our MGCL model and answer the following Research Questions (RQs):

RQ1: How does the proposed model perform compared to baseline models?
RQ2: How do the critical components affect the model performance?
RQ3: How do the different hyperparameter settings affect the performance?
RQ4: How does the proposed model perform under different practical scenarios?

Table 1. Statistics of the four used datasets.

Datasets	Twitter	Douban	Android	Christianity
# Users	12,627	12,232	9958	2897
# Links	309,631	396,580	48,573	35,624
# Cascades	3442	3475	679	589
Avg. Len.	32.60	21.76	33.30	22.90

4.1 Experimental Settings

Datasets. We study the model performance using four real-world social network datasets, i.e. **Twitter** [27], **Douban** [28], **Android** [10], and **Christianity** [10]. These datasets are widely used and each contains the social network and information diffusion cascades. The descriptive statistics are in Table 1.

Evaluation Metrics. According to the problem formulation, the prediction task can be regarded as a retrieval problem. Therefore, we choose the widely used *Hits rate* on top-k (Hits@k) and *Mean Average Precision* on top-k (MAP@k) to evaluate the performance, where k ={10, 50, 100}.

Compared Methods. We compare our method with seven recent information diffusion prediction methods. **Topo-LSTM** [7] and **NDM** [8] utilize RNNs and CNNs to effectively capture the dependencies inherent in the diffusion cascade. **SNIDSA** [9], **Inf-VAE** [10] and **FOREST** [11] employ GNNs to learn users' social relationships and utilize RNN to explore the context of cascades for prediction. **DyHGCN** [12] and **MS-HGAT** [13] focus on integrating various user relationships and then designing novel GNNs for user representation learning.

Implementation Details. The experiments are conducted using PyTorch on a 24GB NVIDIA TITAN RTX GPU. For each dataset, we split these cascades by 8:1:1 for training, validation, and testing. The maximum cascade length is set to 200. To avoid information leakage, we use the training data as the historical cascades. We set the embedding dimension $d = 128$ and the attention head $B = 6$. For the model training, the AdamW optimizer is chosen, initialized with a learning rate of 5e-4 and a weight decay coefficient set to 0.01. The training configuration includes a batch size of 32 and a dropout rate of 0.3. The maximum number of training epochs in the downstream task is set to 50, and we early stop after 3 consecutive epochs without observed improvement on the validation set. The number of training epochs in the graph training is chosen from 4 to 8. The settings for the baselines remain consistent with the original papers.

Table 2. The performance on **Hits**@k metrics (%), scores are the higher the better.

Models	Twitter			Douban			Android			Chiristianity		
	@10	@50	@100	@10	@50	@100	@10	@50	@100	@10	@50	@100
TopoLSTM	8.45	15.80	25.42	8.57	16.53	21.47	4.56	12.63	16.53	12.28	22.63	31.52
NDM	15.21	28.23	32.30	10.00	21.13	30.14	4.85	14.24	18.97	15.41	31.36	45.86
SNIDSA	25.37	36.64	42.89	16.23	27.24	35.59	5.63	15.22	20.93	17.74	34.58	48.76
Inf-VAE	14.85	32.72	45.72	8.94	22.02	35.72	5.98	14.70	20.91	18.38	38.50	51.05
FOREST	28.67	42.07	49.75	19.50	32.03	39.08	9.68	17.73	24.08	24.85	42.01	51.28
DyHGCN	31.88	45.05	52.19	18.71	32.33	39.71	9.10	16.38	23.09	26.62	42.80	52.47
MS-HGAT	33.50	49.59	58.91	21.33	35.25	42.75	10.41	20.31	27.55	28.80	47.14	55.62
MGCL	**34.28**	**51.79**	**62.47**	**21.37**	**35.44**	**43.27**	**10.79**	**21.43**	**29.35**	**31.70**	**50.22**	**60.17**

Table 3. The performance on **MAP**@k metrics (%), scores are the higher the better.

Models	Twitter			Douban			Android			Chiristianity		
	@10	@50	@100	@10	@50	@100	@10	@50	@100	@10	@50	@100
TopoLSTM	8.51	12.68	13.68	6.57	7.53	7.78	3.60	4.05	4.06	7.93	8.67	9.86
NDM	12.41	13.23	14.30	8.24	8.73	9.14	2.01	2.22	2.93	7.41	7.68	7.86
SNIDSA	15.34	16.64	16.89	10.02	11.24	11.59	2.98	3.24	3.97	8.69	8.94	9.72
Inf-VAE	19.80	20.66	21.32	11.02	11.28	12.28	4.82	4.86	5.27	9.25	11.96	12.45
FOREST	19.60	20.21	21.75	11.26	11.84	11.94	5.83	6.17	6.26	14.64	15.45	15.58
DyHGCN	20.87	21.48	21.58	10.61	11.26	11.36	6.09	6.40	6.50	15.64	16.30	16.44
MS-HGAT	22.49	23.17	23.30	11.72	12.52	12.60	6.39	6.87	6.96	17.44	18.27	18.40
MGCL	**22.96**	**23.55**	**23.79**	**11.76**	**12.64**	**12.81**	**6.76**	**7.20**	**7.31**	**18.81**	**19.64**	**19.80**

4.2 Overall Performance (RQ1)

To evaluate the effectiveness of our proposed MGCL, we compare the overall prediction performance of MGCL with the seven recent baselines. The experimental results are summarized in Table 2 and Table 3.

Specifically, we have the following observations: First, compared to the best baseline MS-HGAT, our MGCL adaptively constructs a multiplex heterogeneous social network to mitigate weak structure present in the original social network and conduct the multi-level contrastive learning that can provide more self-supervised signals for graph learning. As a result, MGCL achieves more expressive user representations and consistently outperforms all baseline models in Hits and MAP scores. Second, these methods that incorporate multiple user relationships (DyHGCN, MS-HGAT, and MGCL) show better performance than social-only methods (SNIDSA, FOREST, Inf-VAE), while cascade-only methods (Topo-LSTM, NDM) show the worst performance. This underscores the importance of sufficient social network information for downstream prediction.

4.3 Ablation Study (RQ2)

To investigate the effect of each component on the MGCL, we compare different variants with the original model. Specifically, **w/o GE** only uses the original

social network without enhancing the graph. **w/o GC** replaces adaptive relationships aggregation with average aggregation. Furthermore, we remove our graph training objectives to obtain **w/o** $\mathcal{L}_{con}$, **w/o** $\mathcal{L}_{ucl}$, and **w/o** $\mathcal{L}_{pcl}$.

Table 4. Performance of MGCL with its variants.

Models	Twitter		Android	
	H@100	M@100	H@100	M@100
MGCL	**62.47**	**23.79**	**29.35**	**7.31**
w/o GE	59.82	21.03	28.11	6.97
w/o GC	61.45	20.82	29.18	7.24
w/o $\mathcal{L}_{con}$	60.59	19.23	28.50	7.02
w/o $\mathcal{L}_{ucl}$	61.46	21.12	29.11	7.25
w/o $\mathcal{L}_{pcl}$	60.45	23.01	28.74	7.13

As shown in Table 4, MGCL achieves the best performance compared to any of its variants. Specifically, we can observe the following: First, utilizing the original social network leads to a significant degradation in performance, highlighting the detrimental impact of an incomplete social network on user learning. Second, employing the average aggregation of various user relationships also diminishes performance, underscoring the complexity of relationships. Our structure regularization can ensure that the enhanced structure aligns more closely with real-world scenarios. Third, there is a significant performance degradation upon the removal of either the user-level or preference-level contrastive learning objective, confirming the crucial role of self-supervised signals in graph learning.

4.4 Hyperparameter Analysis (RQ3)

We study four important hyperparameters $\gamma_1, \gamma_2, \gamma_3$ in Eq. (11) and δ in Eq. (7), which determine the effectiveness of the three self-supervised graph learning objectives we proposed in this paper.

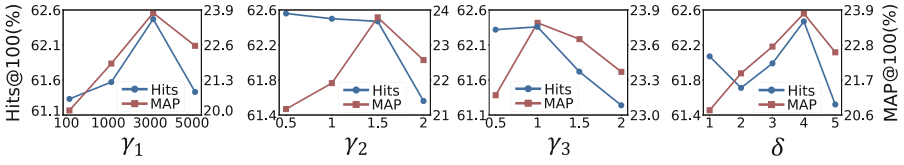


Fig. 3. Impact of different hyperparameters on Twitter.

Figure 3 shows the effect of hyperparameters on the Twitter dataset, the weight γ_1 of the graph regularization is vital to enhance social network structure. The weight γ_2 determines the distance between the user and others, which

is more important for the prediction precision. γ_3 and δ determine the user's preference dependence, and excessively bringing similar users closer will lead to an overly dense distribution of user representations, resulting in suboptimal performance.

4.5 Performance in Different Scenarios (RQ4)

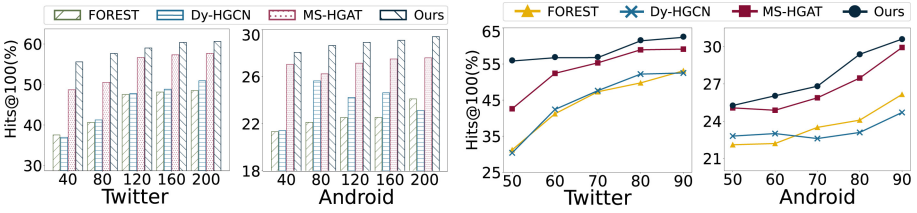
The Performance of High-ranking Position. In some specific scenarios, where precise identification of the evolution of information diffusion is crucial, there is a heightened requirement for more accurate prediction. To address this need, we investigate the performance of high-ranking position prediction.

Table 5. The performance of High-ranking position.

Datasets	Models	H/M@1	H@3	M@3	H@5	M@5
Twitter	MGCL	**15.91**	**24.77**	**19.88**	**28.69**	**20.63**
	MS-HGAT	15.46	23.40	18.95	26.81	19.73
Android	MGCL	**5.38**	**7.30**	**6.12**	**8.31**	**6.42**
	MS-HGAT	5.24	6.94	5.95	7.97	6.18

The results in Table 5 consistently demonstrate that MGCL outperforms MS-HGAT in high-ranking position prediction. We attribute this superiority to MGCL's ability to effectively learn the real behavior of users in social networks. In particular, multi-level contrastive learning of user representations greatly enhances the understanding of users' preferences in actual information diffusion, leading to a substantial improvement in the precision of high-ranking position.

Impact of Cascade Interaction Data. We conduct comparative experiments on Twitter and Android datasets under different training proportions and cascade lengths to further prove the stability of our model.



(a) Maximum length of cascade (b) Proportion of training cascade(%)

Fig. 4. Impact of cascade interaction data.

Referring to Fig. 4(a), we can observe that our model consistently achieves the best performance across different cascade lengths. Notably, for shorter cascade lengths, there is minimal degradation in performance compared to other models. Furthermore, as shown in Fig. 4(b), MGCL achieves comparable performance to the state-of-the-art model trained with less data. In conclusion, these results demonstrate the effectiveness of MGCL learning from user relationships even with extremely weak information issues.

5 Related Work

Graph Contrastive Learning. Graph data, often lacking labels, poses a challenge for current learning methods. Recently, contrastive learning has emerged as a promising solution, which utilizes self-supervised strategies to achieve effective graph learning. DGI [21] extends InfoMax [22] to graphs and maximizes MI between the global graph and local node. GraphCL [23] explores the impact of various combinations of different data augmentations. GCA [24] proposes an adaptive augmentation scheme at both topology and attributes. MVGRL [25] introduces graph diffusion to create another view. However, applying to social networks has certain challenges. For example, inappropriate data augmentation introduces noise and distorts user dependencies, and existing methods lack consideration of specific downstream tasks, limiting their effectiveness in practice.

Information Diffusion Prediction. Information diffusion prediction aims to predict future participants based on the current cascade and relevant knowledge, such as social networks, information content, and user profiles [19,20]. Recent methods focus on learning user representations from sequential or structured cascades using extended RNNs. For example, [7] extends the standard LSTM to model the cross-dependency of cascades. [9] introduces structural information into sequential information using RNN. However, these methods overlook social relationships among users. Some works [10,11] further attempt to embed social relationships to enhance prediction using GNNs. However, learning solely from original social network data is too weak to capture users' relationships. In existing datasets, users lack manual labels, initial user features are random, and the observed user relationships are incomplete. To fill this gap, we innovatively enhance the social network structure and propose multi-level contrastive tasks to obtain better user representations, which can benefit downstream prediction.

6 Conclusion

In this paper, we design a novel and effective two-stage framework named MGCL. Different from existing works, we construct the multiplex heterogeneous social network to enhance the original structure, and further design three self-supervised training objectives to improve graph learning. This method mitigates weak information issues of social network data and enables higher-quality

user representations to improve the performance of the downstream task. Extensive experiments are conducted on four real-world datasets to validate that our MGCL method can consistently outperform state-of-the-art methods.

Acknowledgments. This work was supported by the State Key Laboratory of Communication Content Cognition Funded Project No. A32003, National Natural Science Foundation of China No. U22A2025 and No. 61972275.

References

1. Li, J., Yang, Y., Hu, Q., Wang, X., Gao, H.: Public opinion field effect fusion in representation learning for trending topics diffusion. In: NeurIPS (2023)
2. Sun L., Rao, Y., Wu, L., Zhang, X., Lan, Y., Nazir, A.: Fighting false information from propagation process: a survey. In: ACM Computing Surveys, vol. 55(10), pp. 1–38 (2023)
3. Chen, J., Hoops, S., Marathe, A., Mortveit, H., Lewis, B., Venkatramanan, S., et al.: Effective social network-based allocation of COVID-19 vaccines. In: KDD, pp. 1675–1683 (2022)
4. Broekaert, J. B., La Torre, D., Hafiz, F.: Competing control scenarios in probabilistic SIR epidemics on social-contact networks. In: ArXiv./abs/2108.13714 (2021)
5. Cheng, J., Adamic, L., Dow, P., Kleinberg, J. M., Leskovec, J.: Can cascades be predicted? In: WWW, pp. 925–936 (2014)
6. Gao, S., Ma, J., Chen, Z.: Effective and effortless features for popularity prediction in microblogging network. In: WWW, pp. 269–270 (2014)
7. Wang, J., Zheng, V. W., Liu, Z., Chang, K. C.: Topological recurrent neural network for diffusion prediction. In: ICDM, pp. 475–484 (2017)
8. Yang, C., Sun, M., Liu, H., Han, S., Liu, Z., Luan, H.: Neural diffusion model for microscopic cascade study. TKDE **33**(3), 1128–1139 (2021)
9. Wang, Z., Chen, C., Li, W.: A sequential neural information diffusion model with structure attention. In: CIKM, pp. 1795–1798 (2018)
10. Sankar, A., Zhang, X., Krishnan, A., Han, J.: Inf-VAE: a variational autoencoder framework to integrate homophily and influence in diffusion prediction. In: WSDM, pp. 510–518 (2020)
11. Yang, C., Tang, J., Sun, M., Cui, G., Liu, Z.: Multi-scale information diffusion prediction with reinforced recurrent networks. In: IJCAI, pp. 4033–4039 (2019)
12. Yuan, C., Li, J., Zhou, W., Lu, Y., Zhang, X., Hu, S.: DyHGCN: a dynamic heterogeneous graph convolutional network to learn users' dynamic preferences for information diffusion prediction. In: ECML/PKDD, pp. 347–363 (2020)
13. Sun, L., Rao, Y., Zhang, X., Lan, Y., Yu, S.: MS-HGAT: memory-enhanced sequential hypergraph attention network for information diffusion prediction. In: AAAI, pp. 4156–4164 (2022)
14. Liu, Y., Ding, K., Wang J., Lee, V., Liu, H., Pan, S.: Learning strong graph neural networks with weak information. In: KDD, pp. 1559–1571 (2023)
15. Yu, P., Fu, C., Yu, Y., Huang, C., Zhao, Z., Dong, J.: Multiplex heterogeneous graph convolutional network. In: KDD, pp. 2377–2387 (2022)
16. Hamilton, W. L., Ying, R., Leskovec, J.: Inductive representation learning on large graphs. In: NeurPIS (2017)
17. Vaswani, A., et al.: Attention is all you need. In: NeurIPS, pp. 5998–6008 (2017)

18. Wang, Z., Zhu, Y., Wang, C., Ma, W., Li, B., Yu, J.: Adaptive graph representation learning for next POI recommendation. In: WWW, pp. 393–402 (2023)
19. Zhang, H., Yang, Y., Wang, X., Gao, H., Hu, Q.: MLI: A multi-level inference mechanism for user attributes in social networks. In: TOIS, vol. 41(2), 1–30 (2022)
20. Wang, H., Yang, C., Shi, C.: Neural information diffusion prediction with topic-aware attention network. In: CIKM, pp. 1899–1908 (2021)
21. Velickovic, P., Fedus, W., Hamilton, W.L., Li'o, P., Bengio, Y., Hjelm, R.D.: Deep graph infomax. In: ICLR (Poster) (2019)
22. Hjelm, R.D., et al.: Learning deep representations by mutual information estimation and maximization. In: arXiv preprint arXiv:1808.06670 (2018)
23. You, Y., Chen, T., Sui, Y., Chen, T., Wang, Z., Shen, Y.: Graph contrastive learning with augmentations. In: NeurIPS, pp. 5812–5823 (2020)
24. Zhu, Y., Xu, Y., Yu, F., Liu, Q., Wu, S., Wang, L.: Graph contrastive learning with adaptive augmentation. In: WWW, pp. 2069–2080 (2021)
25. Hassani, K., Khasahmadi, A. H.: Contrastive multi-view representation learning on graphs. In: ICML, pp. 4116–4126 (2022)
26. An, W., Tian, F., Chen, P., Tang, S., Zheng, Q., Wang, Q.: Fine-grained category discovery under coarse-grained supervision with hierarchical weighted self-contrastive learning. In: EMNLP, pp. 1314–1323 (2022)
27. Hodas, N.O., Lerman, K.: The simple rules of social contagion. In: Scientific Reports, pp. 1–7 (2014)
28. Zhong, E., Fan, W., Wang, J., Xiao, L., Li, Y.: ComSoc: adaptive transfer of user behaviors over composite social network. In: KDD, pp. 696–704 (2012)